provided by master PSU firmware **610** within MC **110**. Methods **800** and **900** are described as being implemented via micro-controller **220**A and particularly the execution of code provided by master PSU firmware **650** within micro-controller **220**A. It is however appreciated that certain aspects of the described methods may be implemented via other processing devices and/or execution of other code.

[0054] With specific reference to FIG. **7**, method **700** begins at the start block and proceeds to block **701** where MC **110** triggers PSUs **130**A-C to turn on and initialize after input power has been provided. MC **110** triggers each of the PSUs via communication bus **134** to disable their output on positive Vout terminal **316** (block **702**). At block **704**, MC **110** retrieves the data identifying the master PSU **622** and slave PSUs **624** from MC memory **114**. MC **110** transmits a first control signal identifying one of PSUs as the master PSU (i.e. PSU **130**A) and transmits a second control signal identifying the other PSUs as slave PSUs (i.e. PSUs **130**B and **130**C) (block **705**). MC **110** triggers the master PSU **130**A to turn on MOSFET **240**A and the slave PSUs **130**B and **130**C to turn off MOSFETs **240**B and **240**C respectively (block **706**), such that the master PSU **130**A sets the voltage on the common share bus **250**. MC **110** triggers each of the PSUs to enable their output on positive Vout terminal **316** (block **708**).

[0055] MC **110** receives from each of the PSUs **130**A-C an indicator of the operational condition of each of the PSUs (block **710**). At decision block **712**, MC **110** determines if all of the PSUs have a good operational condition. In response to all of the PSUs having a good operational condition, MC **110** continues to monitor all of the PSUs for good operational condition. In response to at least one of the PSUs **130**A-C not having a good operational condition, MC **110** triggers the defective PSUs to shutdown or be disabled (block **714**) and determines if the defective PSUs include the master PSU (block **716**).

[0056] In response to a defective PSU being the master PSU, MC **110** determines a new master PSU (block **718**) and transmits a control signal signaling the new master PSU that it is the master PSU (block **720**). In one embodiment, MC **110** can randomly select a new master PSU from among the PSUs having a good operational condition. MC **110** triggers the new master PSU to turn on its respective MOSFET **240** (block **722**) such that the new master PSU sets the voltage on the common share bus **250**. After block **722**, at decision block **724**, MC **110** checks if input power has been lost to all of the PSUs **130**A-C within IHS **100**. If input power is still present, MC **110** returns to block **710** and continues receiving the operational condition of the PSUs. If input power has been lost to all of the PSUs, method **700** ends. Method **700** would re-start when input power is re-applied. In response to a defective PSU not being the master PSU, MC **110** continues to monitor whether all of the remaining PSUs have a good operational condition at block **712**.

[0057] Turning to FIG. **8**, method **800** begins at the start block and proceeds to block **802** where micro-controller **220**A receives a control signal from MC **110** indicating whether PSU **130**A is the master PSU or a slave PSU. At decision block **804**, micro-controller **220**A determines if the control signal indicates that PSU **130**A is a master PSU. In response to the control signal indicating that PSU **130**A is a master PSU, micro-controller **220**A turns on MOSFET **240**A (block **806**) such that PSU **130**A is the master PSU that sets the voltage on the common share bus **250**. Method **800** then terminates. In response to the control signal indicating that PSU **130**A is not a master PSU, micro-controller **220**A turns off MOSFET **240**A (block **808**) such that PSU **130**A is a slave PSU. Method **800** then ends.

[0058] Turning to FIG. **9**, method **900** begins at the start block and proceeds to block **902** where micro-controller **220**A initializes PSU **402**A during start-up, including providing a signal via D/A **510** to the common hardware bus **420**. Micro-controller **220**A detects the slot voltage at node **526** via A/D **520** (block **906**) and detects the voltage on the common hardware bus **420** via circuit line **532** and A/D **520** (block **908**). Micro-controller **220**A compares the slot voltage at node **526** to the voltage on the common hardware bus **420** (block **910**). Micro-controller **220**A determines if the slot voltage is less than the voltage on the common hardware bus **420** (decision block **912**).

[0059] In response to the slot voltage not being less than the voltage on the common hardware bus **420**, micro-controller **220**A identifies PSU **402**A as a master PSU and turns on MOSFET **240**A (block **914**) such that PSU **402**A sets the voltage on the common share bus **250**. In response to the slot voltage being less than the voltage on the common hardware bus **420**, micro-controller **220**A identifies PSU **402**A as a slave PSU and turns off MOSFET **240**A (block **916**). Micro-controller **220**A determines if an output voltage is present on positive Vout terminal **316** (decision block **918**). If no output voltage is present on positive Vout terminal **316**, converter **210**A and PSU **402**A are not operating or are operating in a defective manner. In response to determining that an output voltage is present on positive Vout terminal **316**, micro-controller **220**A continues to detect the slot voltage at node **526** via A/D **520** (block **906**). In response to determining that an output voltage is not present on positive Vout terminal **316**, micro-controller **220**A disables the slot voltage signal (block **920**). By disabling the slot voltage signal one of the other PSUs (i.e. PSU **402**B or **402**C) can then become the master PSU. Method **900** then ends.

[0060] In the above described flow charts, one or more of the methods may be embodied in a computer readable medium containing computer readable code such that a series of functional processes are performed when the computer readable code is executed on a computing device. In some implementations, certain steps of the methods are combined, performed simultaneously or in a different order, or perhaps omitted, without deviating from the scope of the disclosure. Thus, while the method blocks are described and illustrated in a particular sequence, use of a specific sequence of functional processes represented by the blocks is not meant to imply any limitations on the disclosure. Changes may be made with regards to the sequence of processes without departing from the scope of the present disclosure. Use of a particular sequence is therefore, not to be taken in a limiting sense, and the scope of the present disclosure is defined only by the appended claims.

[0061] Aspects of the present disclosure are described above with reference to flowchart illustrations and/or block diagrams of methods, apparatus (systems) and computer program products according to embodiments of the disclosure. It will be understood that each block of the flowchart illustrations and/or block diagrams, and combinations of blocks in the flowchart illustrations and/or block diagrams, can be implemented by computer program instructions. Computer program code for carrying out operations for